**3. Report**

*Group Progress*:

For design review 2, we built the 16-bit ADD, SUB, SHIFT, REGISTER and 8:1 MUX top level connections for our design project. For each of these functions, we first built the sub-circuit for one bit location (e.g. a simple two input ADD gate) and then used the saved sub-circuit to build the 16-bit function block. For each transistor level circuit, we used minimum sizing for NMOS and PMOS because there were no requirements for metrics other than functionality for this design review. We will take sizing into account once we finish building the circuit and start to improve its performance. For SUB function block, we used 2’s complement subtraction because it is the easiest to implement in hardware. Since we didn’t talk about handling negative numbers, we left the result in 2’s complement form. For the ADD, we used the basic full adder topology for simplicity. It now consumes a lot of power.

In order to prove that each function works as expected, we simulated each circuit with buffered inputs (double-inverted inputs) and an inverter with transistors four times the minimum size in Cadence as the driven load. The simulation shows that each component works for different combinations of inputs. Because there are 16-bit inputs/outputs for each function block, we decided not to simulate every possible combination of inputs for every bit. Instead, we chose combinations that are representative to show that the block works (e.g. for SHIFT, we showed 1-bit, 2-bit, 3-bit, and 4-bit shifts for a certain 16-bit input).

We also wired up the entire ALU with top level input/output connections in place. The ALU can perform 8 different functions: NOP, ADD, SUB, SHIFT, AND, OR, PASS and MULTIPLICATION (for our arbitrary function). It has inputs A, B and Control. Control is the selection input to the 8:1 MUX, the MUX passes the selected function to perform on A and B. Two inputs and ALUout are 16 bits, while Control is 3-bit wise and CarryOut has only 1 bit.

*Remaining Tasks*:

We have finished building the structure of the ALU block and the registers that will be used in the DSP system except for the arbitrary design. Before the next design review, which is also the final report/presentation due date, we are to finish the design of our arbitrary function block (Multiplier), finish wiring up the ALU and registers, and improve the performance of the DSP to achieve the lowest design metric given by *(Active Power)\*Delay^2\*Area*. Then we will pull components together, finalize the DSP design, and work on our project paper and presentation.

**4. Final Decision on Arbitrary Function**

Multiplication of the lower 8 bits of inputs A and B – approved by Professor Calhoun

**5. Worst Case Delay Table**

|  |  |  |
| --- | --- | --- |
| **Operation** | **Worst-case Delay** | **Scenario** |
| ADD | 948.6 ps | A:0000000000000000->0000000000000001  B:0111111111111111->0111111111111111 |
| SUB | 967 ps | A:0000000000000000->0000000000000000  B:1000000000000001->1000000000000000 |
| SHIFT | 33.2 ps | A:1100101001010011->1100101001010011  B:00->10\* |
| AND | 8.7 ps | A:**1**000000000000000->**1**000000000000000  B:**0**000000000000000->**1**000000000000000\*\* |
| OR | 9.2 ps | A:**0**000000000000000->**0**000000000000000  B:**1**000000000000000->**0**000000000000000\*\* |
| PASS A | 8 ps | A:1111111111111111->0000000000000000 |

*\* We assume that all paths from input through two MUXes are critical except the first three with “0” as one of the inputs*

*\*\*We assume the same worst case delay for every bit location in the 16-bit AND/OR; bit highlighted is the one analyzed*